

WHAT IS CLAIMED IS:

- 1 1. A method of operating a computer system that includes first and second
2 processors and memory shared thereby, the method comprising:
3 concurrently executing first and second instructions on respective ones of the
4 first and second processors, the first and second instructions each
5 reserving in a same predefined order plural respective locations of the
6 memory,
7 wherein, for at least the first instruction, signaling of a fault corresponding to a
8 later reserved one of the respective locations depends on a value read
9 from an earlier reserved one of the respective locations.

- 1 2. The method of claim 1,
2 wherein the predefined order is in accordance with a fixed total order of
3 locations within the memory.

- 1 3. The method of claim 1,
2 wherein the predefined order is one of ascending and descending memory
3 address order.

- 1 4. The method of claim 1,
2 wherein the reserving includes locking an associated cache-line.

- 1 5. The method of claim 1,
2 wherein the reserving locks at least the respective location, but substantially
3 less than all the memory.

- 1 6. The method of claim 1,
2 wherein the first and second instructions are linearizable synchronization
3 operations.

- 1 7. The method of claim 1,
2 wherein the first instruction is a double compare-and-swap instruction.

1 8. The method of claim 1,
2 wherein the first instruction is a compound compare-and-swap instruction;
3 wherein the respective locations reserved by the compound compare-and-swap
4 instruction number N; and
5 wherein signaling of a fault corresponding to a later reserved one of N
6 locations depends on at least one value read from an earlier reserved
7 one the N locations.

1 9. The method of claim 1,
2 wherein the first instruction is a double compare-and-swap instruction; and
3 wherein, unless a value read from the earlier reserved one of the respective
4 locations compares to a corresponding test value, no fault
5 corresponding to the later reserved one of the respective locations is
6 signaled.

1 10. The method of claim 1,
2 wherein the execution of the first instruction includes access to the earlier
3 reserved one of the respective locations; and
4 wherein, unless the access succeeds, no fault corresponding to the later
5 reserved one of the respective locations is signaled.

1 11. The method of claim 1, wherein the execution of the first instruction
2 includes
3 a first access corresponding to an earlier reserved one of the respective
4 locations; and
5 an optional second access that signals a fault only if the first access succeeds.

1 12. The method of claim 1,
2 wherein the respective locations reserved by the first and second instructions
3 are disjoint; and
4 wherein the concurrent execution is non-blocking.

1 13. A processor that implements an instruction that separately reserves plural
2 memory locations in an order prescribed by a fixed total order of the memory
3 locations.

1 14. The processor of claim 13,
2 wherein the fixed total order of the memory locations is one of ascending
3 memory address order and descending memory address order.

1 15. The processor of claim 13,
2 wherein the separately reserved memory locations include an earlier reserved
3 first memory location and a later reserved second memory location;
4 and
5 wherein signaling of a fault corresponding to the second memory location
6 depends on a value read from the first memory location.

1 16. The processor of claim 13,
2 wherein the instruction is a compound compare-and-swap instruction; and
3 wherein the instruction signals a fault corresponding to a later reserved one of
4 the memory locations only if a value read from an earlier reserved one
5 of the memory locations compares to a test value.

1 17. The processor of claim 13,
2 wherein the instruction signals a fault corresponding to a later reserved one of
3 the memory locations only if access to an earlier reserved one of the
4 memory locations succeeds.

1 18. The processor of claim 13, wherein access to at least the first memory
2 location is one of:
3 a compare-and-swap access;
4 a test-and-set access;
5 a read-compute-conditional write access; and
6 a read-modify-write access.

1 19. The processor of claim 13,
2 wherein the reserving includes locking an associated cache line.

1 20. A computer system comprising:
2 first and second processors and storage shared thereby;
3 the first and second processors each implementing an instruction that
4 separately reserves locations of the storage addressed thereby in an
5 order prescribed by a fixed total order of the locations.

1 21. The computer system of claim 20,
2 wherein, on execution of the instruction, the locations addressed thereby
3 include an earlier reserved first location and a later reserved second
4 location; and
5 wherein signaling of a fault corresponding to the second location depends on a
6 value read from the first location.

1 22. The computer system of claim 20,
2 wherein the instruction is a compound compare-and-swap operation; and
3 wherein the instruction signals a fault corresponding to a later reserved one of
4 the locations only if a value read from an earlier reserved one of the
5 locations compares to a test value.

1 23. The computer system of claim 20,
2 wherein instances of the instruction that address disjoint sets of the locations
3 are concurrently executed by the first and second processors.

1 24. The computer system of claim 20, further comprising:
2 a coherently maintained set of caches including first and second caches
3 respectively associated with the first and second processors,
4 wherein, if a first instance of the instruction and a second instance of the
5 instruction address locations associated with distinct sets of cache
6 lines, then the first and second instances of the instruction are
7 concurrently executable by the first and second processors.

1 25. A computer program product encoded in at least one computer readable
2 medium, the computer program product comprising:
3 a first set of instructions executable on a first processor,
4 the first set including at least one instance of a particular instruction directing
5 the processor to separately reserve plural storage locations referenced
6 thereby in a predefined order such that concurrent execution of the
7 particular instruction and a corresponding instruction avoids deadlock,
8 the corresponding instruction also separately reserving plural storage
9 locations referenced thereby in the predefined order.

1 26. The computer program product of claim 25,
2 wherein the particular instruction itself provides reservation in the predefined
3 order.

1 27. The computer program product of claim 25,
2 wherein the first set includes additional instructions to ensure that storage
3 location references supplied to the particular instruction are ordered in
4 accordance with the predefined order.

1 28. The computer program product of claim 25, further comprising:
2 a second set of instructions executable on a second processor,
3 the second set including at least one instance of the corresponding instruction.

1 29. The computer program product of claim 25,
2 wherein the predefined order is ascending memory address order.

1 30. The computer program product of claim 25,
2 wherein the predefined order is descending memory address order.

1 31. The computer program product of claim 25,
2 wherein the particular instruction and the corresponding instruction are same
3 instructions.

1 32. The computer program product of claim 25,
2 wherein the particular instruction and the corresponding instruction each
3 implement a compound Compare-and-swap (nCAS) operation.

1 33. The computer program product of claim 25,
2 wherein the at least one computer readable medium is selected from the set of
3 a disk, tape or other magnetic, optical, or electronic storage medium
4 and a network, wireline, wireless or other communications medium.

1 34. A computer system that allows two or more DCAS instructions to operate
2 concurrently if they operate on distinct cache lines.

1 35. A processor that implements a first instruction that reserves, in an
2 predefined order, plural storage locations referenced thereby, wherein concurrent
3 execution of the first instruction and a corresponding instruction that also reserves, in
4 the same predefined order, plural storage locations referenced thereby is non-blocking
5 if the first and the corresponding instructions reference distinct memory portions.

1 36. The processor of claim 35, further implementing the corresponding
2 instruction.

1 37. The processor of claim 35,
2 wherein the storage locations are sharable with a second processor; and
3 wherein the corresponding instruction is implemented at least by the second
4 processor.

1 38. The processor of claim 35,
2 wherein the reserving includes locking the referenced storage locations.

1 39. The processor of claim 35,
2 wherein the reserving includes locking cache lines associated with the
3 referenced storage locations.

1 40. A processor of claim 35,
2 wherein the predefined order is ascending memory address order.

1 41. A processor of claim 35,
2 wherein the predefined order is descending memory address order.

1 42. The processor of claim 35,
2 wherein the first instruction and the corresponding instruction are same
3 instructions.

1 43. The processor of claim 35,
2 wherein the first instruction and the corresponding instruction both implement
3 a compound Compare-and-swap (nCAS) operation.

1 44. An apparatus comprising:
2 a memory store; and
3 means for separately reserving in response to a single instruction, plural
4 locations of the memory store in a predefined order in accordance with
5 a fixed total order of locations in the memory store.

1 45. The apparatus of claim 44, further comprising:
2 a cache,
3 wherein the means for separately reserving includes means for separately
4 locking cache lines associated with each of the plural locations.

1 46. The apparatus of claim 44, further comprising:
2 means for signaling, if at all, a fault corresponding to a later reserved one of
3 the locations based on a result of access to an earlier reserved one of
4 the locations.

1 47. A method of operating a computer system that includes a memory shared
2 by plural processors thereof, the method comprising:
3 in response to execution of a single instruction by one of the processors,

4 separately reserving plural locations of the memory; and
5 signaling a fault corresponding to a later reserved one of the locations
6 based on a value read from an earlier reserved one of the
7 locations.

1 48. The method of claim 47,
2 wherein, unless the value read from the earlier reserved location compares to a
3 corresponding test value, no fault corresponding to the later reserved
4 location is signaled.

1 49. The method of claim 47,
2 wherein the separately reserving includes separately locking at least the plural
3 locations, but substantially less than all the memory.

1 50. The method of claim 47,
2 wherein the computer system further includes cache storage; and
3 wherein the separately reserving includes separately locking respective cache
4 lines associated with the plural locations.

1 51. The method of claim 50,
2 wherein the cache storage includes a coherently maintained set of caches
3 including ones respectively associated with each of the plural
4 processors.

1 52. The method of claim 47,
2 wherein the instruction implements a compound Compare-and-swap (nCAS)
3 operation.

1 53. The method of claim 47,
2 wherein the instruction implements a Double Compare-and-swap (DCAS)
3 operation.

1 54. The method of claim 47,
2 wherein at least one of the plural locations is identified by an operand of the
3 instruction.

1 55. A method of operating a processor, the method comprising:
2 in response to execution by the processor of a single instruction that separately
3 accesses plural memory locations,
4 accessing a first one of the memory locations; and
5 signaling a fault corresponding to a second one of the memory
6 locations depending on a result of the first memory location
7 access.

1 56. The method of claim 55, further comprising:
2 in response to the execution of the single instruction,
3 reserving the first and second memory locations;
4 detecting the fault as part of the second memory location reservation;
5 and
6 signaling the fault, if at all, only upon success of the first memory
7 location access.

1 57. The method of claim 55,
2 wherein the first memory location access includes comparing a value read
3 from the first memory location to a corresponding test value; and
4 wherein the signaling is performed, if at all, only upon success of the
5 comparing.

1 58. The method of claim 55, further comprising:
2 reserving the first and second memory locations; and
3 signaling the fault corresponding to the second memory location, if at all,
4 based on a value read from the reserved first memory location.

1 59. A processor that implements an instruction that addresses first and second
2 memory locations but for which signaling of a fault corresponding to the second
3 memory location depends on a value read from the first memory location.

1 60. The processor of claim 59,
2 wherein, unless the value read compares to a test value, no fault corresponding
3 to the second memory location is signaled.

1 61. The processor of claim 59,
2 wherein the instruction is a compound compare-and-swap instruction.

1 62. The processor of claim 59,
2 wherein the instruction separately reserves the first and second memory
3 locations;
4 wherein the fault, if any, corresponding to the second memory location is
5 detected upon reservation thereof, but signaled, if at all, based on the
6 value read from the reserved first memory location.

1 63. A computer program product encoded in at least one computer readable
2 medium, the computer program product comprising:
3 a set of instructions executable on a processor,
4 the set of instructions including at least one instance of an instruction directing
5 the processor to access first and second memory locations but for
6 which signaling of a fault corresponding to the second memory
7 location depends on a value read from the first memory location.

1 64. The computer program product of claim 63,
2 wherein the instruction further directs the processor to separately reserve the
3 first and second memory locations; and
4 wherein the fault, if any, corresponding to the second memory location is
5 detected upon reservation thereof, but signaled, if at all, based on the
6 value read from the reserved first memory location.

1 65. The computer program product of claim 63,
2 wherein the instruction further directs the processor to reserve the first and
3 second memory locations in a predefined order in accordance with a
4 fixed total order of memory locations.

1 66. The computer program product of claim 63,
2 wherein, unless the value read compares to a test value, no fault corresponding
3 to the second memory location is signaled.

1 67. The computer program product of claim 63,
2 wherein the instruction is a compound compare-and-swap instruction.

1 68. The computer program product of claim 63,
2 wherein the at least one computer readable medium is selected from the set of
3 a disk, tape or other magnetic, optical, or electronic storage medium
4 and a network, wireline, wireless or other communications medium.

1 69. An apparatus comprising:
2 a memory store;
3 means for accessing in response to a single instruction, first and second
4 locations of the memory store; and
5 means for signaling, if at all, a fault corresponding to the second location
6 based on a value read from the first location.